

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Currently Amended) A computer-readable storage device storing a computer program product for deriving a metadata API from a metamodel in order to develop an application, the computer program product being operable to cause data processing apparatus to:

receive the metamodel in a first language, the metamodel describing a diagram of classes that define the one or more development objects, the development objects representing building blocks for developing the application;

convert the metamodel to a model description that describes the metamodel in a second language according to an interchange format;

generate a set of intermediate objects to represent the classes of the metamodel by parsing the model description; and

generate code ~~that is included in the API~~ using the set of intermediate objects as inputs[,] ~~wherein the~~ to derive an API enabl[~~(es)~~]ing development tools to access the development objects to develop the application.

2. (Cancelled).

3. (Previously Presented) The computer-readable storage device of claim 1, wherein the second language comprises XML.

4. (Previously Presented) The computer-readable storage device of claim 1, wherein the first language comprises UML.

5. (Cancelled).

6. (Previously Presented) The computer-readable storage device of claim 1, wherein the first language comprises a customizable extension.

7. (Previously Presented) The computer-readable storage device of claim 6, wherein the customizable extension is used to implement an additional feature of the API.

8. (Previously Presented) The computer-readable storage device of claim 7, wherein the additional feature comprises an indication of a file border.

9. (Previously Presented) The computer-readable storage device of claim 1, wherein the API comprises a copy and paste operation.

10. (Currently Amended) A computer-readable storage device storing a computer program product for deriving a metadata API from a metamodel in order to

develop an application, the computer program product being operable to cause data processing apparatus to:

receive the metamodel in a first language, the metamodel describing a diagram of classes that define ~~the~~ one or more development objects, the development objects representing building blocks for developing the application, wherein the first language comprises unified modeling language;

convert the metamodel to a model description that describes the metamodel in a second language according to an interchange format, wherein the second language comprises XML;

generate a set of intermediate objects to represent the classes of the metamodel by parsing the model description; and

generate code ~~that is included in the API~~ using the set of intermediate objects as inputs~~[[,]] wherein the~~ to derive an API includ~~[[es]]ing~~ an XML schema that enables implementation of the development objects, and further wherein the API enables development tools to access the development objects to develop the application.

11. (Cancelled).

12. (Previously Presented) The computer-readable storage device of claim 10, wherein the second language comprises XML.

13. (Cancelled).

14. (Previously Presented) The computer-readable storage device of claim 10, wherein the set of intermediate objects comprises Java objects.

15. (Previously Presented) The computer-readable storage device of claim 10, wherein the XML schema includes a tree based on aggregation relationships in the metamodel.

16. (Previously Presented) The computer-readable storage device of claim 10, wherein the XML schema includes a reference based on an association relationship in the metamodel first model.

17. (Previously Presented) The computer-readable storage device of claim 10, wherein the XML schema includes a complex type extension based on an inheritance relationship in the metamodel.

18. (Currently Amended) A computer-readable storage device storing a computer program product for deriving metadata API from a metamodel in order to develop an application, the computer program product being operable to cause data processing apparatus to:

receive the metamodel describing a diagram of classes that define ~~the~~ one or more development objects, the development objects representing building blocks for developing the application;

generate an XML model that is a representation of the metamodel according to an interchange format;

generate a set of intermediate objects to represent the classes of the metamodel by parsing the XML model using an XML parser; and

generate code ~~that is included in the API~~ using the set of intermediate objects as inputs[[,]] ~~wherein the~~ to derive an API enabl[[es]]ing development tools to access the development objects to develop the application.

19. (Cancelled).

20. (Previously Presented) The computer-readable storage device of claim 18, wherein the operations comprise:

creating a new development object as a transient object without an existing corresponding file; and

modifying the transient object until the transient object is committed to a persistent file.

21. (Previously Presented) The computer-readable storage device of claim 20, further comprising instructions to destroy the transient object if a delete command is requested before the transient object is committed to a persistent file.

22. (Previously Presented) The computer-readable storage device of claim 20, further comprising instructions to mark the persistent file as deleted if a delete command is requested after the transient object is committed to a persistent file.

23. (Previously Presented) The computer program product of claim 1, wherein the metamodel is stored on one of a storage module, a server, and a portable storage device.

24. (Currently Amended) The ~~computer program product~~ computer-readable storage device of claim 18, wherein the metamodel is stored on one of a storage module, a server, and a portable storage device.

25. (Previously Presented) The computer-readable storage device of claim 1, wherein the set of intermediate objects comprises Java objects.

26. (Previously Presented) The computer-readable storage device of claim 18, wherein the set of intermediate objects comprises Java objects.